Automated Centerline for Computed Tomography Colonography[1]

Gheorghe Iordanescu, PhD, Ronald M. Summers, MD, PhD

Rationale and Objectives. A novel method to compute the centerline of the human colon obtained from computed tomography colonography is proposed. Two applications of this method are demonstrated: to compute local colonic distension (caliber), and to match polyps on supine and prone images.

Materials and Methods. The centerline algorithm involves multiple steps including simplification of the colonic surface by decimation; thinning of the decimated colon to create a preliminary centerline; selection of equally spaced points on the preliminary centerline; grouping neighboring points; and mapping them back to rings in the original colon. This method was tested on 20 human computed tomography colonography datasets (supine and prone examinations of 10 patients) and on a computer-generated colon phantom.

Results. Visual inspection of the colons and their centerlines showed the centerline to be accurate. For the colon phantom, the average error was only 1 mm. For 11 polyps visualized in both the supine and prone positions and found by computer-aided detection, the normalized distance along the centerline to each polyp was not significantly different on the supine and prone views ($r = 0.999$; $P < .001$).

Conclusion. This method produces an accurate colon centerline that may be useful for flight path planning, matching detections on the supine and prone views, and computing local colonic distension.

Key Words. Computed tomography (CT); colon; computed tomography; 3D reconstruction; centerline detection; colon cancer; image processing.

Computed tomography colonography (CTC, also known as virtual colonoscopy) is a relatively new diagnostic test for the colon. It consists of a computed tomography (CT) scan of the patient rather than the traditional invasive colonoscopic inspection. With CTC, one can diagnose colon cancer or polyps at an early stage (1).

Recently, several research groups have developed computer-aided detection (CAD) software that identifies colonic polyps (2). The purpose of CAD is ultimately to improve the performance and lower the cost of CTC. Computed-aided detection is in an early stage of development. Early experiments indicate that a number of important techniques may play roles in improving CAD, including recognizing normal structures such as the ileocecal valve, assessing and responding to differing degrees of colonic distension or caliber, and using the supine and prone views together (3). To this end, we report a novel centerline algorithm for CTC that: (A) creates a coordinate system for use in recognizing different parts of the colon; (B) assesses local colonic distension; and (C) allows matching of polyp detections on the supine and prone views.

Acad Radiol 2003; 10:1291–1301

[1] From the Department of Radiology, National Institutes of Health, Building 10, Room 1C660, 10 Center Dr, MSC 1182, Bethesda, MD 20892-1182. Received June 5, 2003; revision requested July 13; revision received August 8; revision accepted August 12 . Presented at the SPIE 2003 Medical Imaging Conference, February 2003, San Diego, CA. **Address correspondence to** R.M.S.

doi:10.1016/S1076-6332(03)00464-1

MATERIALS AND METHODS

Given the 3-dimensional (3D) surface of a colon acquired from a CTC examination, we sought to find an ordered set of 3D points that define the colon's centerline. We first provide a brief overview of the implementation and then give more details of the algorithm.

Our method involves six steps (decimation, thinning, modeling, remapping, centerline computation, and final remapping) applied sequentially (Fig 1). The starting point is the original colon surface (S_{OC}), produced from the CTC images using a region growing segmentation and isosurface extraction (4). The threshold for the isosurface extraction was −800 Hounsfield unit (HU). The "marching cubes" algorithm was used to extract the isosurface (5). The surface is composed of many small triangles, which are described by their vertices and edges. In this article, vertices are described using the notation V[i].

Decimation (Figure 1–Step 1)

The S_{OC} is simplified using decimation to minimize the number of operations performed in subsequent steps. Decimation keeps the general appearance and topology of the colon but reduces the number of vertices by approximately 80%; it does this by removing vertices in regions of the surface with low curvature (6,7). The result of this step is the decimated colon surface (S_{DC}). We used the visual toolkit (VTK; Kitware Inc, Clifton Park, NY) implementation of decimation (6,7) and chose the following parameters required by that software: maximum number of iterations, 200; target reduction, 0.1; initial error 0.003; preserve topology, ON; preserve edges, OFF; boundary vertex deletion, ON.

Thinning (Figure 1–Step 2)

We computed the thinned colon surface (S_{TC}) by iteratively averaging the distances between colon vertices in the S_{DC} (Fig 2). The thinning is equivalent to applying a Laplacian operator to each vertex V[i] (8). If vertex V[i] has N[i] neighbors and all of these neighbors are in a neighborhood N_i, the formula for thinning is:

$$V[i] = \frac{\sum_{v[j] \in N_i} V[j]}{N[i]} \quad (1)$$

Thinning is done iteratively; we use 1,000 iterations (Table 1). The main effect of applying the Laplacian operator is a shrinkage of the local colon diameter. There are two side effects of thinning: tight loops are smoothed and the entire colon is compressed slightly. Both side effects are corrected in the remapping step. Figure 2 shows the consequences of differing numbers of thinning iterations. We can see that the local average diameter of the colon is much smaller and the local folds are collapsed. The number of vertices does not change by thinning. S_{DC} and S_{TC} have the same number of vertices and

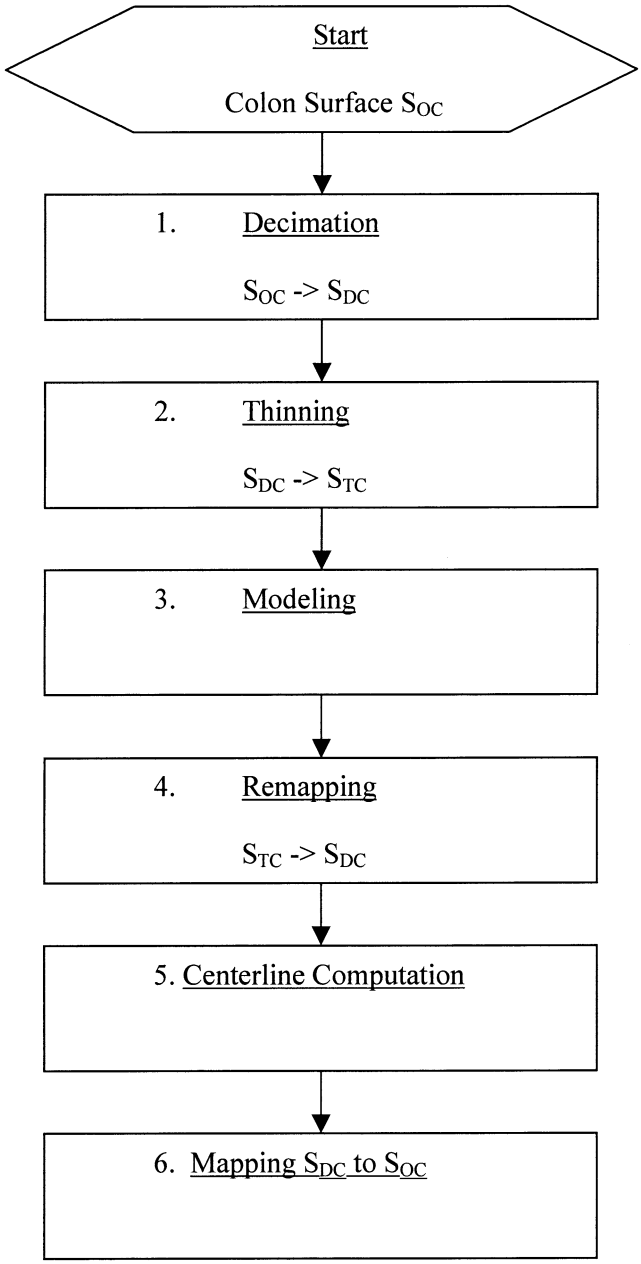


Figure 1. Flowchart of the centerline computation algorithm.

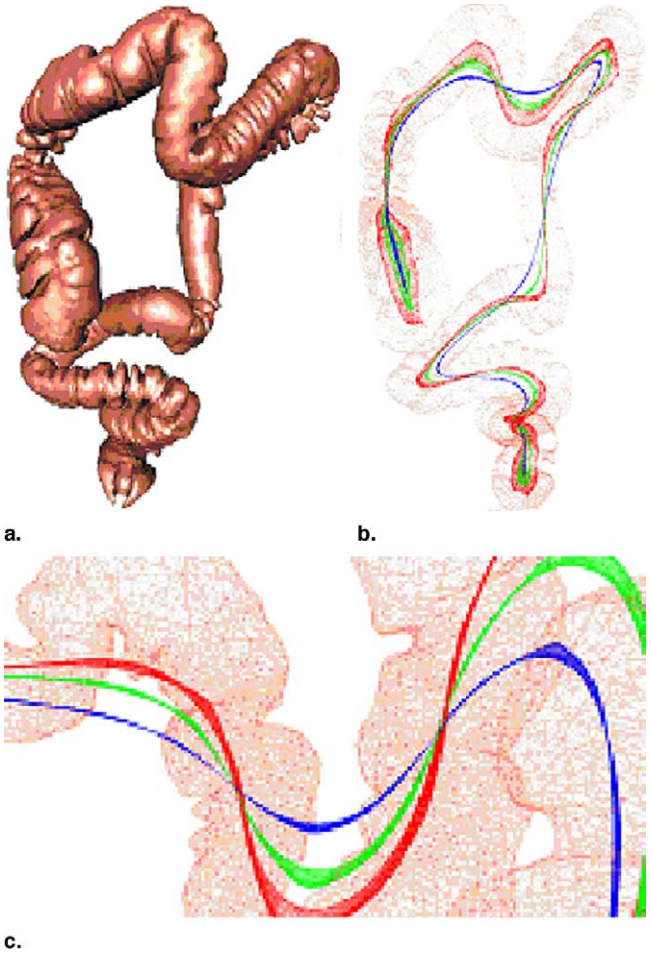


Figure 2. Thinning evolution of a 3D colon surface (step 2 of Fig 1). **(a)** Three-dimensional surface of the human colon reconstructed from a CT colonography dataset. **(b)** Thinned 3D surface of the human colon; 250 (dark red), 500 (green), and 1,000 (blue) iterations. **(c)** Detail.

the same vertex connectivity. We are using floating point numbers to store and process the coordinates of the vertices so the vertices can move as close as desired. The movement of vertices starts to decrease as the vertices become close because the average of a vertex's neighbors will be close to the vertex's present position. By reducing the number of vertices and increasing the mean spacing between vertices, decimation improves thinning efficiency.

Table 1
Centerline Computation Parameters

Description	Value
Number of thinning iterations (Figure 1-step 2)	1,000
Modelling distance (Figure 1-step 3)	1 (cm)
Thickness of ring edges used to compute the centerline point (Figure 1-step 5)	0.015 of the colon span along the z-axis (approximately 6 mm)
Size of the cube used for vertex classification (Figure 1-step 6)	1 (cm)

Modeling (Figure 1–Step 3)—We create a model of the S_{TC} by taking equally spaced vertices from the S_{TC} (Fig 3). The model of the S_{TC} will be composed of straight segments that connect these approximately equally spaced vertices from the S_{TC}. To obtain the correct vertices, we use a region growing strategy in which we start from a seed point and identify vertices connected to the seed.

Let V[i] be vertex index i from the S_{TC} and N_i be all neighbors of V[i]. MD is the modeling distance, 1 cm, the average distance between modeling vertices (Table 1). The following pseudocode visits each vertex, stores in a stack indices requiring further processing and saves indices of the S_{TC} model vertices in a vector called MI.

Pseudocode:

Create and initialize to zero table "Visited," having length equal to the number of vertices in S_{TC};

Create array MI, having 0 length;

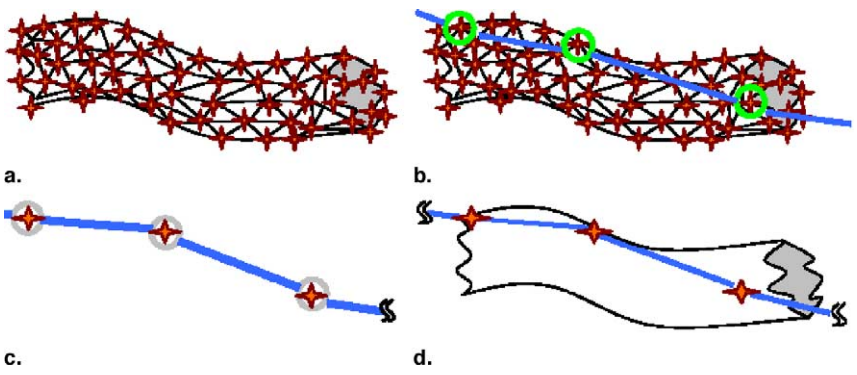


Figure 3. Modeling the colon by an ordered set of 3D points (step 3 of Fig 1). **(a)** Portion of the thinned colon. Note that the diameter of this portion is much less than that of the original colonic segment from which it was derived. Its vertices (red crosses) and their connectivity are displayed. **(b)** Some vertices are selected to model the thinned colon (green circles). **(c)** Model of the thinned colon. The model is now a piece-wise linear curve instead of a surface mesh. **(d)** Portion of the thinned colon and its piece-wise linear model.

```
Select a random index r;
Append r to MI;
CurrentModelIndex = r;
Visited[r] = 1;
Push r onto the stack;
While (stack is not empty){
  int i = Pop from stack;
  Compute N_i;
  For each V[k] in N_i {
    If (Visited[k] != 1){
      If ((Distance between V[CurrentModelIndex]
    and V[k]) > MD){
        Append CurrentModelIndex to MI;
        CurrentModelIndex = k;
      }
      Push V[k] onto the stack;
      Visited[k] = 1;
    }
  }
}
```

When the above loop stops MI will contain the indices of the vertices selected to model the S_{TC}. We only push unvisited indices in the stack so the above loop is guaranteed to stop. Because MD is several times bigger than the local diameter of the S_{TC}, none of the direct neighbors of initial vertex V[r] will pass the distance criterion. By checking its neighbors and moving from neighbor to neighbor (and marking each visited vertex) we will ultimately find another vertex V[r1] at distance MD from V[r] (thus, MI[1] will be set to r1). We subsequently add vertex V[r2] at distance MD from V[r1] and so on. When the above loop ends, the last model index e (the last vertex index appended to MI) is close to the end of the S_{TC} and we perform a final step by searching the farthest vertex V[f] from vertex e in the direction (V[e]–V[e-1]). We then append f to MI and thus obtain the complete model, with the last ring being shorter that the average but being sure we covered the whole S_{TC}.

The resultant model covers the entire colon under the condition that the original vertex V[r] is at either end of the colon (rectum or cecum). But because r is a random number, we cannot guarantee that this condition will hold, so we need to modify our algorithm. Therefore, we compute two models of the S_{TC}, one covering from V[r] to the rectum, the other from V[r] to the cecum, and then merge the two models.

For purposes of display, the ordered set of 3D points is spline-interpolated to provide a smoother curve by adding three new points between each two existing model points. Figure 4 shows the results of thinning and modeling.

Remapping (Figure 1 – Step 4)

We use the model to compute slices through the S_{TC} that correspond to rings in the S_{DC}. Rings are cross-sections of the S_{DC} that are roughly small cylinders about 1 cm in height. Using the one-to-one mapping of the vertices on the S_{TC} and S_{DC}, the indices of the vertices from the same slice in the S_{TC} are used to get a ring in the S_{DC}. The algorithm that performs the S_{TC} - S_{DC} remapping is:

For each 3D point P_i of the model:

1. Take adjacent points P_i and P_{i+1} of the model.
2. Compute two planes perpendicular to the S_{TC} model, one passing through each of the corresponding points P_i and P_{i+1}. These planes define a perpendicular *ring* of the S_{TC}. The plane through P_i is perpendicular to the segment defined by P_{i-1} and P_{i+1}, while the plane through $P_{i+1}$is perpendicular to the segment defined by P_i and P_{i+2}.
3. Select all the vertices of the S_{TC} that lie between these two planes.
4. Map these vertices one-to-one to the S_{DC}.

The result is vertices from the S_{DC} that are grouped in ring-like areas, where each ring is approximately perpendicular to the colon centerline. Because the colon is a deformable structure, the rings normally appear irregular as shown in Fig 4.

Centerline Computation (Figure 1–Step 5)

The centers-of-mass of the edges of adjacent rings of the S_{DC} is the local centerline point. We determine the local centerline point by averaging the vertices at the rings' edges. The edge of a ring is defined as 0.015 of the colon amplitude on the Z direction (6 mm for a 40 cm span along the Z axis; Table 1). The resulting points at the center of each ring are again interpolated for purposes of display using spline functions. The ring centers and interpolated points constitute the final centerline. For interpolation we use the Catmull-Rom spline (9).

Approximately 2% of the interpolated points may fall outside the colonic lumen. The majority of these occur because of the distortion of the lumenal surface caused by the presence of the rectal tube, or because the centerline connects disconnected segments of colon secondary to collapsed intervening colonic segments. Less than 1% of the interpolated points fall outside the colonic lumen be-

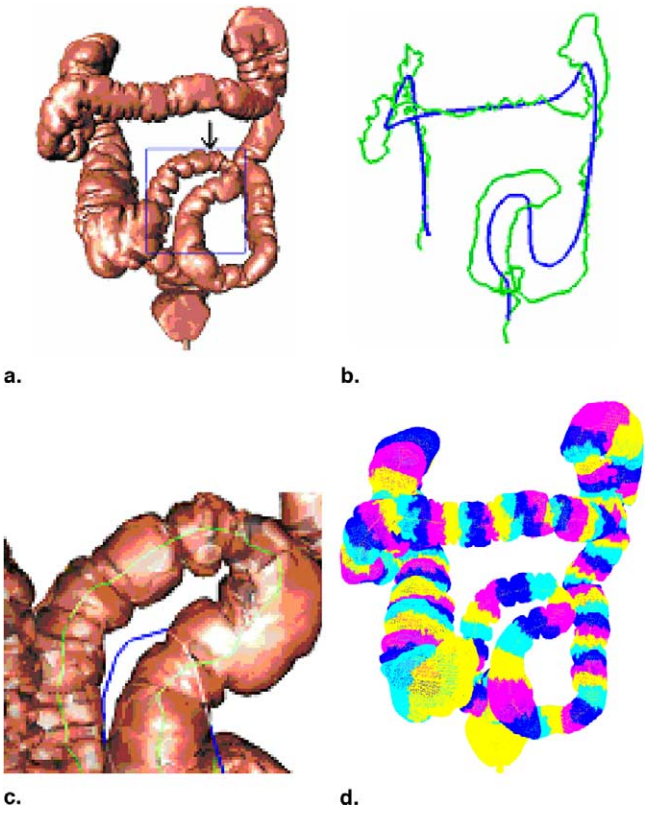


Figure 4. Steps in the centerline algorithm. **(a)** Original 3D colonic surface from prone CTC. **(b)** Blue curve shows the preliminary centerline after thinning and modeling. Preliminary centerlines may not lie within the colonic lumen. Green curve shows the final centerline after remapping. The final centerline lies within the colonic lumen. **(c)** Detail of sigmoid colon (blue box in **(a)**). **(d)** Segmented colon. Colon rings (cross-sections) are colored.

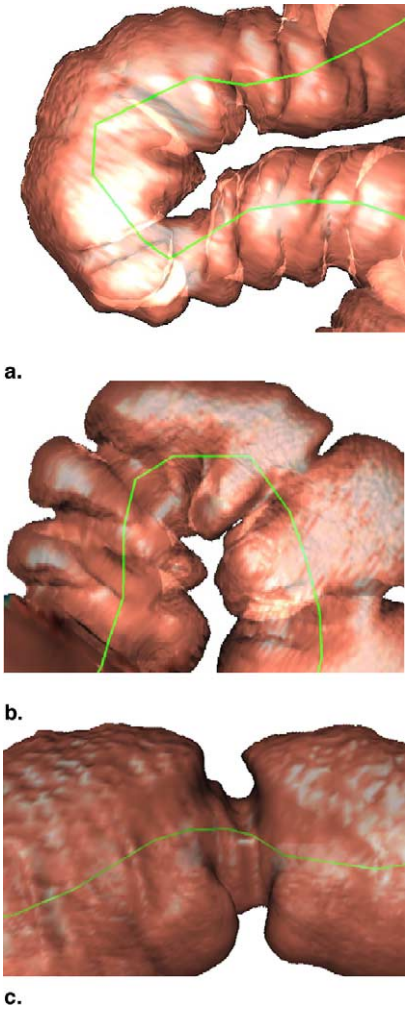


Figure 5. Detail of the centerline in different colonic segments. Colonic surfaces are semitransparent with the corresponding centerlines shown as a green curve. This is the same patient shown in Figure 4. Portions of the **(a)** splenic flexure, **(b)** hepatic flexure, and **(c)** transverse colon are shown.

cause of other causes, for example, if the ring contains a large concavity, which may occur when the colon is partially collapsed. In this case, a heuristic is applied. First, interpolated points falling outside the colon (V_0) are detected by determining intersections between the centerline and the colonic wall. Second, for each point V_0, a plane is computed which contains V_0 and is normal to the vector formed by $V_1 - V_{-1}$, where V_1 and V_{-1} are points adjacent to V_0 on the centerline. In this plane a series of 10 rays are computed that pass through V_0 and cover the entire plane. Each ray is traversed until it pierces the colonic surface twice. Finally, we keep the ray that gives the smallest distance between V_0 and the colon surface. The centerline point V_0 is then moved along the ray to the midpoint of the two ray-surface intersections. In rare cases, a portion of the centerline may lie outside the colonic surface, even though the points on either end of the line segment may lie within the lumen. This situation may be addressed by adding additional interpolating points along the line and then adjusting the line as above. The heuristic described here is relatively computationally expensive (approximately 90 seconds) and is generally unnecessary for practical applications. Therefore, we do not include timing for this step in the reported results. The results shown in this article were obtained without this correction.

Mapping the S_{DC} to the S_{OC} (Figure 1–Step 6)

A second mapper associates vertices in the S_{DC} with the vertices in the S_{OC} based on a minimum distance criterion between the vertices of the two surfaces. The correspondence is one to many so that a vertex from the S_{DC} may be mapped to one or more vertices in the S_{OC}. Based

on the correspondence of the vertices on the S_{DC} and S_{OC} we can segment the S_{OC} and split the surface into rings (Fig 4). The centerline of the S_{OC} is the same as the centerline of the S_{DC}.

To limit the search space and improve computational efficiency in performing this second mapping, a process called "vertex classification" is performed. Vertices in the S_{DC} and $S_{OC}$are grouped into classes corresponding with 1 cm (Table 1) cubes according to their spatial coordinates. All vertices in the same cubic centimeter of volume are assigned to the same class. Because only vertices in neighboring classes need to be searched, there is a substantial performance improvement, provided each class has the same number of vertices. So, for example, if vertices were evenly distributed throughout a $40 \times 40 \times 40$ cm volume, the speed improvement would be greater than 1,000. In practice, the classes have differing numbers of vertices so the acceleration will be variable.

Smoothed Centerline for Correlation with Optical Colonoscopy Distances

The computed centerlines could be very twisted (following every distortion of the colon) and could overestimate the length of a typical colon. By choosing a thicker ring (larger modeling distance) and thinner edge (Table 1), we obtain smoother centerlines and more appropriate colon length values on the order of 1.5 meters. We recognize that this calibration step injects arbitrariness into the method and definition of colon length. However, it should be recognized that because the colon is a distensible and stretchable structure, it does not have one best length, only an approximate length.

Patient Scanning

Computed tomography colonography was performed on 10 patients in both supine and prone positions yielding 20 CTC datasets with same-day conventional colonoscopic validation. Nine patients were scanned using a multi-detector helical CT scanner (10) and one was scanned using a single-detector helical CT scanner (11). The acquisition technique consisted of 5 mm collimation, reconstruction interval 3 mm, 120 kVp and 20–120 mA. We corrected for the scan overlap for the one patient scanned on the single-detector scanner. Seven of the patients had 11 colonoscopically proven polyps (mean size at colonoscopy 1.4 cm; range, 0.3–2.6 cm) seen on both the supine and prone views. These polyps were detected on both views by a CAD system previously described (4,12).

Validation

We tested the accuracy of the centerline in three ways. First, we built a colon phantom composed of four disjoint cylinders, each of radius 1.0 cm, meant to simulate straight and angulated colonic segments. The phantom is a digital construct created entirely in software. We computed the error between 100 centerline points and corresponding points along the axis of the cylindrical phantom.

Second, we evaluated the accuracy of the centerlines of the patient data. Unfortunately, there is no accepted definition of a perfect centerline, so for this part of the validation, our goal was a centerline that stayed inside the colon and that was roughly in the "center" of the colon. To ascertain whether we had met this goal, we evaluated the accuracy of the centerline by visual inspection, with particular attention to how the centerline behaved in sharply angulated or partially collapsed colonic segments. We used an opaque surface reconstruction of the colon to confirm that the centerline stayed inside the colon. Also, by increasing the transparency of the surface, we were able to compare the centerline and the original colon simultaneously. We also ran a computer program that analyzed the centerlines to detect centerline points that lie outside the colonic lumen.

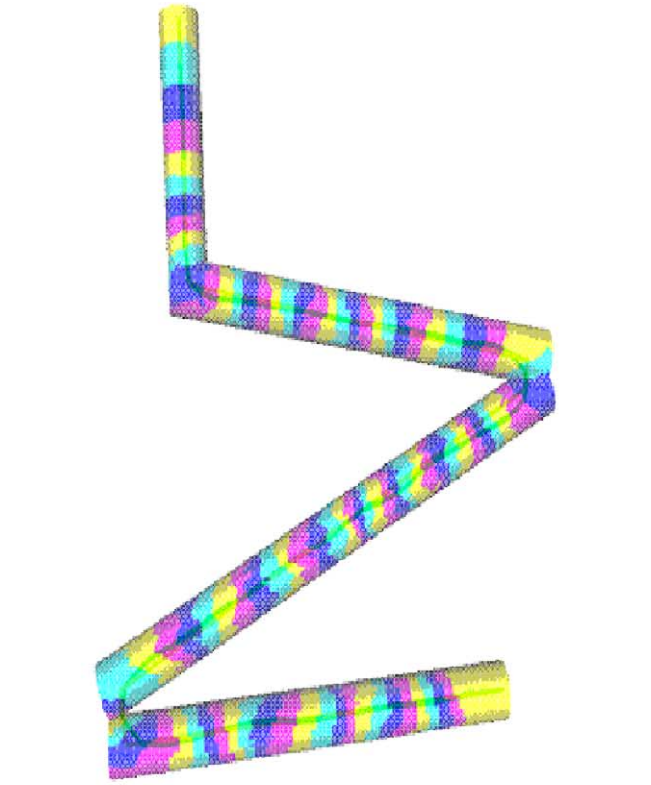
Third, we evaluated the accuracy of the coordinate system that is produced as a byproduct of the centerline algorithm. To perform this validation, we determined whether the location of the internal landmarks (the polyps described above), computed using the centerline, agreed on the supine and prone views.

Implementation Details

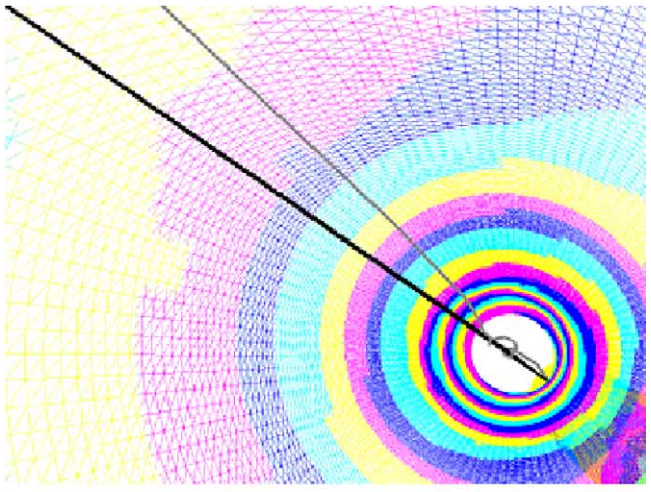
Our algorithm is written in C++ using the Open Inventor API (Silicon Graphics, Inc, Mountain View, CA), the Qt cross-platform GUI (Trolltech, Oslo, Norway) and the SoQt library (Systems in Motion AS, Oslo, Norway). The programs were run on a Dell Workstation PWS530 PC (Dell Computer Corp., Round Rock, TX) with Intel 1.5 GHz processor, 400 MHz system bus, 2 GB memory (RAM) and Microsoft Windows 2000 operating system (Microsoft Corp., Redmond, WA).

Applications

The utility of the centerline was shown for two applications. First, the volume of each ring was used to quantify local colonic distension. Second, the normalized distance along the centerline (NDAC) was computed and compared for a series of polyps seen on both the prone and supine examinations.



a.



b.

Figure 6. Simple colonic phantom. **(a)** Centerline (green) and colon rings (colored cross-sections) are shown. **(b)** Detail from inside colon phantom. The black straight line is the cylinder's axis and the grey curve is the computed centerline.

Paired *t*-tests were used to compare lengths and distensions on the supine and prone views. In the figures, to correct a mirror inversion in the CAD software, images have been flipped left-to-right to place the colons in their correct anatomic position.

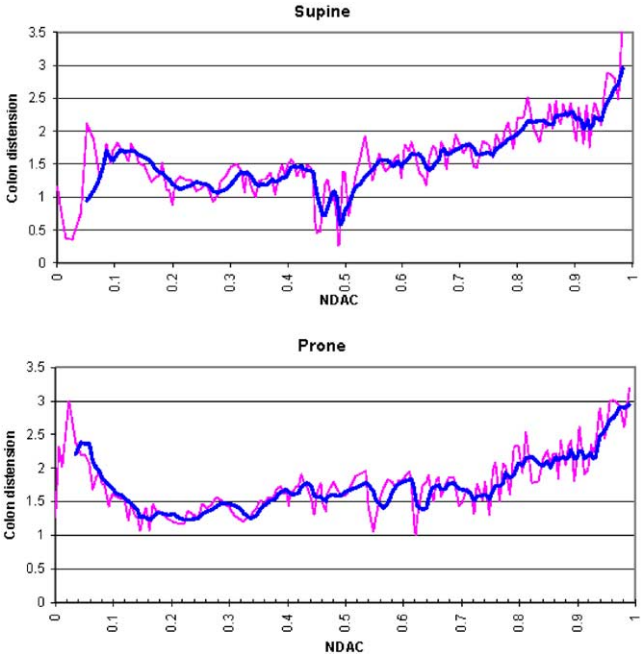


Figure 7. Distension along the length of the colon for the supine and prone scans of the same patient shown in Figure 4. The plots show local colon profiles as a thin pink curve and rolling averages in dark blue. The rolling averages were performed with a mobile window of five samples to provide a smoother curve and show the local trend. The horizontal axis values are the normalized distance along the centerline (NDAC) for each ring (0 = rectum, 1 = cecum). The vertical axis values are the average distension of the colon ring, expressed as the mean distance of the colonic wall from the centerline in centimeters. Because these distension values are in some sense radial measurements, they correspond with approximately half the visually apparent colonic distension that might be measured by a radiologist.

RESULTS

We obtained a visually appropriate centerline in all 20 cases. Example centerlines are shown in Figures 4 and 5.

For the digital colon phantom, the computed centerline was also excellent. The average error for the colon phantom was 1.1 ± 1.0 mm, which is appropriate for clinical use (Fig 6).

Figure 7 shows the distension of one patient's colon on the supine and prone views plotted as a function of position along the centerline. Although the two colon profiles in Figure 7 may look visually similar, this is not a typical result. Usually colon profiles are not similar in the supine and prone positions because of high variability of the colon and different scanning conditions (air is typically lost and re-insufflated between the scans). However the final colon lengths (198 and 195 cm, respectively) and average distension (1.6 and 1.8 cm, respectively) were

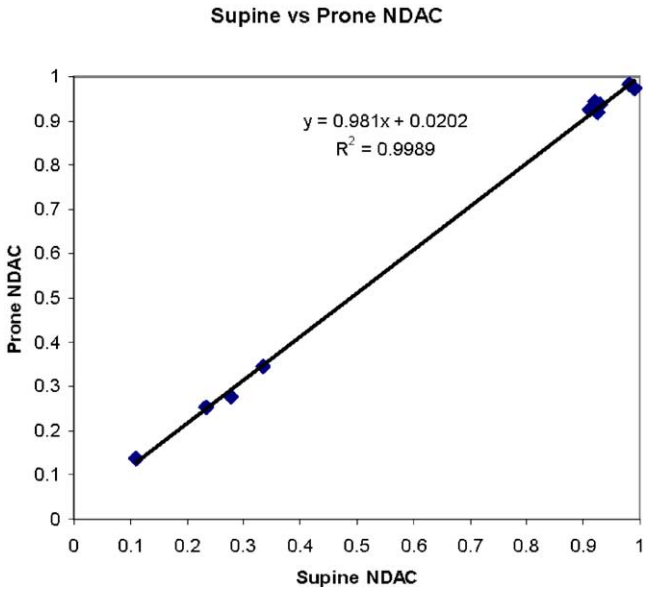


Figure 8. Comparison of normalized distance along the centerline (NDAC) for the supine and prone CT colonography exams for 11 polyps in seven patients. Each polyp was found by computer-aided detection on both the supine and prone examinations. There is excellent correlation ($r^2 = 0.999$) between the NDAC values indicating an accurate match.

very similar, as expected. Also, as could be expected from normal anatomy, the rectum and cecum (0 and 1 on the X axis in the figure, respectively) had the greatest distension and the rectum was better distended on the prone exam.

The mean colon lengths (± SD) for the supine and prone exams were 146 ± 26 and 144 ± 28 cm, respectively (Table 2). The mean signed difference was −7 ± 9 cm (NS, $P = .67$).

Table 2
Centerline Computation Results

	Supine	Prone	*P* Value
Colon length (cm)	146 ± 26	144 ± 28	.67
Distension (cm)	1.6 ± 0.2	1.5 ± 0.2	.28
Colon rings (n)	119 ± 22	120 ± 26	.92

NOTE. Values on supine and prone scans are expressed as mean ± standard deviation. There are no statistically significant differences between the measured values on the supine and prone scans.

The distensions for the supine and prone exams, expressed as the average distance of points on the wall of the colon as measured from their corresponding centerline points, were 1.6 ± 0.2 and 1.5 ± 0.2 cm, respectively (NS, p=0.28). Note that each ring contributes one distension value to this calculation, that value being the average distension measured in that ring. Because no rings are present in a collapsed segment, this calculation excludes collapsed segments and provides distension assessment only for partially distended or fully distended colonic segments. The mean numbers of colonic rings for the supine and prone exams were 119 ± 22 and 120 ± 26, respectively (NS, p=0.92).

Figure 8 shows a comparison of normalized distance along the centerline (NDAC) for 11 polyps in 7 supine and prone cases. The slope in Figure 8 is not significantly different from 1 (r=0.999; p<0.001), indicating strong

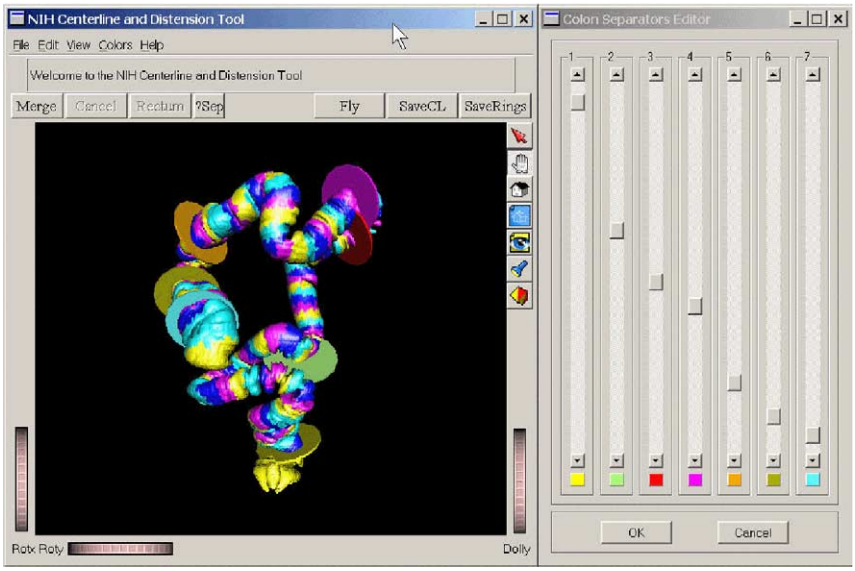


Figure 9. Graphical user interface to display results. The colored disks superimposed on the colonic surface **(left)** indicate the boundary of each anatomic colonic segment, and are placed interactively by moving the sliders **(shown at right)**. There are seven sliders, which separate the eight colonic segments. The disks move along the centerline as the sliders are dragged. When the user has finished placing the disks, the centerline coordinates of the separators are saved to a file. These coordinates enable the conversion of centerline normalized distances to anatomic segments reported by the colonoscopist. The converted distances are useful for automated matching of computer-aided polyp detections with the location of known polyps and for assessing the spatial distribution and dependence on distension of false positive detections.

correlation. The mean signed difference of NDAC on the supine view compared to the prone view was -0.01 ± 0.01 (range -0.03 to 0.02). The "limits of agreement" for the mean signed difference (average difference $\pm$ 2 standard deviations) is -0.03–0.02 (13). For a colon of length 150 cm, the limits of agreement are -5 to 3 cm.

The graphical user interface we developed to enable practical use of the centerline information is shown in Figure 9. The interface displays the colon rings, permits automated fly-throughs, and allows one to define the boundaries between anatomic segments (such as the sigmoid and rectum, or the splenic flexure and descending colon).

The average time for decimation was 45 seconds (maximum, 65 seconds). The average computing time (excluding decimation) was 59 seconds (maximum, 73 seconds). If we include times for decimation we get a total average execution time of 108 seconds for the entire program.

DISCUSSION

The centerline of an anatomic structure is an important tool for medical 3D data analysis and display. The centerline fulfills the need for an anatomically based coordinate system. Such a coordinate system is useful for making measurements, for locating abnormalities, and for guiding intervention. For CTC in particular, an accurate centerline allows the user to perform automated fly-throughs (14–18), to create alternative displays like "virtual pathology" (19), to define anatomic regions (eg, rectum, cecum, etc), to extract important features like distension, and to make statistical assessments such as polyp spatial distribution.

Our algorithm produced a visually accurate centerline for all the colons in our study. The algorithm also provided estimates of local colonic distension and a coordinate system for measuring distance along the centerline from a landmark (eg, the distal rectum). On the phantom, the mean error was only 1.1 mm, which is not significant for the intended use.

We found extremely strong correlation ($r = 0.999$) between the NDAC for the identical polyps detected by CAD on both the prone and supine views. The mean difference in NDAC computed along the supine and prone colon centerlines was only 1%, with a standard deviation of 1%, a difference so small as to be clinically unimportant. In addition, the limits of agreement are from -5 to 3 cm. Given the elasticity of the colon, this range is remarkably good. This result not only validates the accuracy of the centerline coordinates, but indicates a novel application, namely that the NDAC could be used to match polyp detections on the two views. Such matching could improve radiologist efficiency during image interpretation or increase the sensitivity of CAD. The distances along the centerline might also be useful for guiding colonoscopists or surgeons to a target lesion, or in matching detections on follow-up scans, although these applications were not specifically tested.

Our algorithm is fully automated for well-distended colons and fast (less than 2 minutes on a PC). The first and last points of the centerline are also determined automatically.

Because we obtain not only the centerline but also the rings of the colonic surface, we have developed a new method to segment topologically cylindrical shapes. Our method can be used in general in image segmentation of tubular structures in 3D. In addition, our algorithm describes a means to group surface elements of such a structure according to a local coordinate system and hence to enable more efficient processing of such a surface.

Our centerline can also be used to create automated fly-throughs. The centerline provides the virtual fly-path elements: the projection of the rings onto a plane perpendicular to their axis defines what the user wants to see during virtual navigation for inspection of the colonic wall. In other published research, the fly-through is the main application of the centerline (15,17,18,20–25). We did not perform a detailed assessment of fly-throughs using our centerline because CAD, not fly-throughs, is our primary application of the centerline.

The centerline we have developed may be useful for clinical applications other than the colon. For example, it may be useful for other nonbranching anatomic structures such as the esophagus, small bowel, and ureters. One may be able to use the centerline and distension measurements to detect caliber change or lobulation. To segment long tubular structures, one may analyze the orientation or curvature of the centerline to detect bends or tortuosity to, for example, identify the different segments of the duodenum.

In some special situations that occur clinically (eg, collapsed colons) we applied our algorithm for each piece of the colon and then merged the resulting centerline. In such cases we need user supervision to check merging accuracy because it is not always true that the closest ends of two colon pieces are connected. Dealing with

collapsed colons needs to be improved, but is a problem with most published centerline algorithms.

The concept of the centerline of a 3D surface is not well defined. It is close to the more mathematically elaborated concept of skeletons. Intuitively, the centerline of a colon-like surface should be the curve that stays inside the colon and never goes "too close" to the walls.

In addition to natural questions like how to define wall closeness, there are also specific clinical problems. In our case, we desire a reference line to compute distances and distensions along the colon because our purpose is to be able to identify polyps, to compare our detections with colonoscopist-provided data, and to register detections on the supine and prone examinations. Colons differ from patient to patient, for the same patient in supine and prone position (because of gravity and because air may be lost and re-insufflated between inspections), or for the same patient in the same positions but inspected at different times. The colonoscope is semi-rigid and cannot make sharp bends; it will press against some of the colon walls particularly in curving portions such as the flexures and sigmoid colon; and the colon will tend to telescope (bunch up or stretch out) along its length. So, to achieve a centerline that approximates the colonoscope path, we need some sort of smoothed centerline and we allow it to be reasonably close to the wall of the colon. Thus, the colon lengths reported using any method must be regarded as approximate values. Despite these problems, not only were the colon lengths on the supine and prone views not significantly different, but also the calculated differences were so small as to be clinically unimportant.

Centerlines can be used to improve user interaction. One can use the centerline to minimize computational effort during rotations and translations of complex surfaces and volumes. As the 3D surface is moving, the user does not need a high-resolution surface but only an idea about its spatial position. The centerline is the ideal surrogate to the complete surface or volume during such movements. If the centerline is replaced with the original shape once the user stops moving, one obtains a faster user-interactivity without losing static image accuracy.

Other methods of centerline determination include computing center of mass for growing voxels (20), topologic thinning ("onion peeling"), cell decomposition (26), skeletons and cost functions (21,27–29), graph extraction from a distance map (30), Dijkstra shortest path (31), and penalized distance from a source field (15,18,19,22–24). The general approach is to use a regularly spaced discrete grid to place the control voxels of the centerline. To generate an accurate centerline, we need a very fine grid. Thus, we need to process a large number of grid points to decide which belong to the centerline. Our method does not use a grid and our control points can have any position in space. Our method allows flexibility in choice of the control points' positions and consequently requires fewer computational and storage resources.

Topologic thinning is proved to give accurate results but may be computationally expensive and extra care must be taken to keep the topology of the object (15,22). Distance mapping techniques (24,27,31) are faster, but they require more correcting steps because the colon topology is not always preserved: if the colon is thick, it is possible to obtain a branched skeleton. In contrast, our method preserves the colon topology (no branches are added) and is relatively fast (all data processing takes less than 2 minutes). Using the ring's average to compute a centerline point and average colon distension to estimate the colon profile has been proposed as a method to register patient colons on supine and prone CTC (25).

Another important difference between existing methods and ours is that we do not require the whole 3D set of voxels but only the 3D colonic surface. This is advantageous because the 3D colonic surface is already determined by existing CAD software (11). Because of this characteristic, our algorithm may have application to other problem domains in which the input data is a surface. Another benefit is that the colon rings are obtained naturally as part of the algorithm, This is an advantage over other methods that require additional processing steps (32). It may be possible to view polyps more advantageously by slicing the colon into rings and viewing the rings individually.

Our algorithm does have limitations. Because our approach does not use voxels arranged on a rectangular 3D grid, it is more difficult to implement classical morphologic and minimizing cost function algorithms. However, because our algorithm is separable, hybrid approaches to this problem are possible (33). In addition, our algorithm does not deal with topologic handles that can form when parts of the colon touch (27). We are investigating ways to correct these handles using a manual correction but an automated method would be desirable. The algorithm that corrects portions of the centerline that lie just outside the colon is computationally expensive; this problem is uncommon and the correction is unnecessary for most practical applications. The phantom also has limitations because it lacks folds and gentle undulations. However, the

centerline of a more complex structure can be more difficult to define as we have already described.

In summary, we developed and validated a novel centerline algorithm that is fast, fully automated, requires only the colonic surface representation, and provides colonic distension assessment and accurate polyp matching. The algorithm may be useful as part of a CAD system or as an aid to clinical interpretation.

ACKNOWLEDGMENTS

We thank Andrew Dwyer, MD for critical review of the manuscript and C. Daniel Johnson, MD, Mayo Clinic, for providing CT colonography data. We also thank Juan R. Cebral, PhD, George Mason University, for suggesting the use of thinning to obtain a first estimate of the colon centerline.

REFERENCES

1. Johnson CD, Dachman AH. CT colonography: the next colon screening examination? Radiology 2000; 216:331–341.
2. Summers RM, Yoshida H. Future directions: computer-aided diagnosis. In: Dachman AH, ed. Atlas of virtual colonoscopy. New York, NY: Springer, 2003; 55–62.
3. Summers RM. Challenges for computer-aided diagnosis for CT colonography. Abdom Radiol 2002; 27:268–274.
4. Summers RM, Beaulieu CF, Pusanik LM, et al. Automated polyp detector for CT colonography: feasibility study. Radiology 2000; 216:284–290.
5. Lorensen WE, Cline HE. Marching cubes: a high resolution 3D surface reconstruction algorithm. ACM Computer Graphics 1987; 21:163–169.
6. Schroeder WJ, Zarge JA, Lorensen WE. Decimation of triangle meshes. ACM Comp Graphics 1992; 26:65–70.
7. Kitware Inc : The Visualization ToolKit (VTK). Available at: http://kitware.com/vtk/index.html
8. Taubin G. Curve and surface smoothing without shrinkage. Proceedings of the Fifth International Conference on Computer Vision 1995; 852–857.
9. Catmull E, Rom R. A class of local interpolation splines. New York, NY: Academic Press, 1974.
10. Summers RM, Jerebko AK, Franaszek M, Malley JD, Johnson CD. Complementary role of computer-aided detection of colonic polyps with CT colonography. Radiology 2002; 225:391–399.
11. Summers RM, Johnson CD, Pusanik LM, Malley JD, Youssef AM, Reed JE. Automated polyp detection at CT colonography: feasibility assessment in a human population. Radiology 2001; 219:51–59.
12. Summers RM, Jerebko A, Franaszek M, Malley JD, Johnson CD. Complementary role of computer-aided detection of colonic polyps with CT colonography. Radiology 2002; 225:391–399.
13. Bland JM, Altman DG. Statistical methods for assessing agreement between two methods of clinical measurement. Lancet 1986; 1:307–310.
14. Lorensen WE, Jolesz FA, Kikinis R. The exploration of cross-sectional data with a virtual endoscope. In: Satava R, Morgan K, eds. Interactive technology and the new medical paradigm for health care. Washington, DC: IOS Press, 1995; 221–230.
15. Paik DS, Beaulieu CF, Jeffrey RB, Rubin GD, Napel S. Automated flight path planning for virtual endoscopy. Medical Physics 1998; 25: 629–637.
16. Summers RM. Navigational aids for real-time virtual bronchoscopy. AJR Am J Roentgenol 1997; 168:1165–1170.
17. Taosong He LH, Dongqing C, Zhengrong L. Reliable path for virtual endoscopy: ensuring complete examination of human organs. Trans Visual Comput Graph 2001; 7:333–342.
18. Chiou RCH, Kaufman AE, Liang ZR, Hong LC, Achniotou M. An interactive fly-path planning using potential fields and cell decomposition for virtual endoscopy. IEEE Trans Nucl Sci 1999; 46:1045–1049.
19. McFarland EG, Wang G, Brink JA, Balfe DM, Heiken JP, Vannier MW. Spiral computed tomographic colonography: determination of the central axis and digital unraveling of the colon. Acad Radiol 1997; 4:367–373.
20. Samara Y, Fiebich M, Dachman AH, Doi K, Hoffmann KR. Automated centerline tracking of the human colon. Proc SPIE 1998; 3338:740–746.
21. Jie Wang YG. An optimization problem in virtual endoscopy. Theor Comput Sci 1998; 207:203–216.
22. Ge Y, Stelts DR, Wang J, Vining DJ. Computing the centerline of a colon: a robust and efficient method based on 3D skeletons. J Comput Assist Tomogr 1999; 23:786–794.
23. Ge Y, Stelts DR, Zha X, Vining D. Computing the central path of a colon from CT images. Proceedings of the SPIE's International Symposium on Medical Imaging: 1998: Imaging Processing, SPIE 3338; part 1: 702-713. San Diego, California, February 1998.
24. Bitter I, Kaufman AE, Sato M. Penalized-distance volumetric skeleton algorithm. IEEE Trans Visual Comput Graph 2001; 7:195–206.
25. Delphine Nain SH, Grimson WEL, Cosman E Jr, Wells WW, Ji H, Ron Kikinis, Westin CF. Intra-patient prone to supine colon registration for synchronized virtual colonoscopy. MICCAI 2002; LNCS 2002; 2489: 573–580.
26. Hong L, Muraki S, Kaufman A, Bartz D, He T. Virtual voyage: interactive navigation in the human colon. Proceedings of SIGGRAPH '97. Conference, 1997: 27–35.
27. Wan M, Liang ZR, Ke Q, Hong LC, Bitter I, Kaufman A. Automatic centerline extraction for virtual colonoscopy. IEEE Trans Med Imag 2002; 21:1450–1460.
28. Yong Zhou AWT. Efficient skeletonization of volumetric objects. Trans Visual Comput Graph 1999; 5:196–209.
29. Truyen R, Deschamps T, Cohen LD. Clinical evaluation of an automatic path tracker for virtual colonoscopy. Medical Image Computing and Computer-Assisted Intervention (MICCAI) October 2001, Utrecht, The Netherlands.
30. Frimmel H, Nappi J, Yoshida H. Fast and robust method to compute colon centerline in CT colonography. In: SPIE Medical Imaging. San Diego: SPIE, 2003; 5031:381–387.
31. Bitter I, Bender M, McDonnell KT, Kaufman A, Wan M. CEASAR: A smooth, accurate and robust centerline extraction algorithm. IEEE Visualization: Proceedings of the Conference on Visualization 2000, Salt Lake City, Utah, United States, 2000:45-52.
32. Hung PW, Paik DS, Napel S, Jeffrey RB, Yee J, Beaulieu CF. Comparison of computerized methods for quantifying bowel distension in CT colonography using three different metrics: validation in colon phantoms. Radiology Society of North American 86th Scientific Sessions, Chicago, IL, November 2000. Radiology 2000; 217:485.
33. Schirmacher H, Zöckler M, Stalling D, Hege H-C. Boundary surface shrinking - a continuous approach to 3D centerline extraction. In Girod B, Niemann H, Seidel H-P, eds. Proceedings of IMDSP '98, Alpbach, Austria, July 1998:25-28.